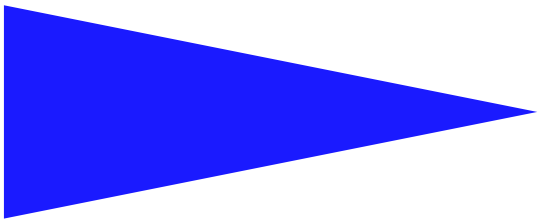
PUBLICATION
INTERNE
N° 1752



NONSTATIONARY CONSISTENCY OF SUBSPACE METHODS

ALBERT BENVENISTE AND LAURENT MEVEL

ISSN 1166-8687

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTÈMES ALÉATOIRES
Campus de Beaulieu – 35042 Rennes Cedex – France
Tél. : (33) 02 99 84 71 00 – Fax : (33) 02 99 84 71 71
http://www.irisa.fr

Nonstationary consistency of subspace methods

Albert Benveniste and Laurent Mevel *

Systèmes numériques
Projet Sisthem

Publication interne n ° 1752 — Octobre 2005 — 30 pages

Abstract: In this paper we study "nonstationary consistency" of subspace methods for eigenstructure identification, *i.e.,* the ability of subspace algorithms to converge to the true eigenstructure despite nonstationarities in the excitation and measurement noises. Note that such nonstationarities may result in having time-varying zeros for the underlying system, so the problem is nontrivial. In particular, likelihood and prediction error related methods do not ensure consistency under such situation, because estimation of poles and estimation of zeros are tightly coupled. We show in turn that subspace methods ensure such consistency. Our study carefully separates statistical from non-statistical arguments, therefore enlightening the role of statistical assumptions in this story.

Key-words: subspace methods, non stationary excitation, martingales

(Résumé : tsvp)

* email: firstname.lastname@inria.fr
This work was supported by the FliTE Eureka project number 2419.

Centre National de la Recherche Scientifique
(UMR 6074) Université de Rennes 1 – Insa de Rennes

Institut National de Recherche en Informatique
et en Automatique – unité de recherche de Rennes

Consistence des méthodes sous espaces sous excitation non stationnaire

Résumé : Dans ce rapport, nous étudions la convergence des méthodes sous espaces dans le cadre de l'identification des structures mécaniques en vibration sous des conditions d'excitation non stationnaire, et plus particulièrement la capacité des algorithmes de sous espaces à converger vers le vrai modèle de structure propre malgré la présence de non stationnarités. De telles non stationnarités peuvent conduire à un système sous-jacent avec des zéros variant dans le temps, le problème est donc non trivial. En particulier, les méthodes de vraisemblance ou basées sur l'erreur de prédiction ne garantissent pas la consistence dans ce cas, parce que l'estimation des zéros et des pôles est couplée. Nous montrons que les méthodes sous espaces garantissent la consistence. Notre étude sépare les arguments statistiques de ceux qui ne le sont pas, de manière à éclairer l'apport des hypothèses statistiques.

Mots clés : Méthodes sous espaces, non stationarité, martingales

1 Introduction

In this paper we study "nonstationary consistency" of subspace methods for eigenstructure identification, *i.e.,* the ability of subspace algorithms to converge to the true eigenstructure despite nonstationarities in the excitation and measurement noises. Note that such nonstationarities may result in having time-varying zeros for the underlying system, so the problem is nontrivial. In particular, likelihood and prediction error related methods do not ensure consistency under such situation, because estimation of poles and estimation of zeros are tightly coupled.

In 1985, Benveniste and Fuchs [6] proved that the Instrumental Variable method and what was called the Balanced Realization method for linear system eigenstructure identification are consistent for the class of nonstationary systems we discuss here. Since this paper, the family of subspace algorithms has been invented [16, 22, 25, 26, 27] and has expanded rapidly. Therefore, we felt it was timely revisiting the results of [6] and generalizing them to subspace methods. To this end, [6] had first to be restructured to show up an important intermediate result, which had not been noticed explicitly in the original paper but was clearly there. Still, the generalization we present here is far less trivial than expected and required introducing new techniques for the proof.

There are a number of convergence studies on subspace methods in a stationary context in the literature, see [13, 2, 3, 4, 10, 11] to mention just a few of them. These papers provide deep and technically difficult results including convergence rates. They typically address the problem of identifying the system matrices or the transfer matrix, *i.e.,* both the pole and zero parts of the system. In contrast, the *nonstationary* consistency property that we study here holds for the estimation of the eigenstructure (the pole part) only and does not apply to the zero part, at least as far as the transfer from unobserved inputs to output measurements is concerned. It is definitely different from the problem considered in [24].

The paper is organized as follows. The problem of nonstationary consistency is stated in Section 2, where a generic form of subspace algorithm is also stated. Section 3 collects the key steps of our analysis; section 3.1 collects the non-probabilistic arguments of the consistency proof; probabilistic arguments of the proof are collected in Sub-section 3.2; and our assumptions are discussed in section 3.3. Finally, in Section 4, by using the so developed toolbox of theorems and lemmas, we prove nonstationary consistency of some representative subspace algorithms.

2 Problem setting, a generic subspace algorithm

Problem setting. Consider the following linear system

$$\left\{ \begin{array}{rcl} x_k & = & Ax_{k-1} + Bu_k + v_k \\ y_k & = & Cx_{k-1} + Du_k + w_k \end{array} \right. \tag{1}$$

where $k \in \mathbb{Z}$, x is the $\mathbb{R}^n$-valued state, u is the $\mathbb{R}^m$-valued observed input, v and w are unobserved input disturbances, and y is the $\mathbb{R}^q$-valued observed output.

The key point of this work is that the unobserved input disturbances can be *nonstationary*. For instance, they can be white noises having unknown time-varying covariance matrices. For this case, we should rather reformulate system (1) in the following form, which enlightens that y_k itself is nonstationary in a nontrivial way:

$$\left\{ \begin{array}{rcl} x_k & = & Ax_{k-1} + Bu_k + K(k)\nu_k \\ y_k & = & Cx_{k-1} + Du_k + L(k)\nu_k \end{array} \right. \tag{2}$$

where

$$\left[\begin{array}{c} K(k) \\ L(k) \end{array} \right] \left[\begin{array}{cc} K^T(k) & L^T(k) \end{array} \right]$$

is the time-varying covariance matrix of the excitation noise in (1), and ν_k is a stationary standard white noise. Note that the zero part of the transfer $\nu_k \mapsto y_k$ is time-varying in this case, so that consistency makes sense only w.r.t. the pole part.

The problem we consider is the *identification of the pair* (C, A) *up to a change of basis in the state space of system* (2). Equivalently, we identify the pairs $(\lambda, C\varphi_\lambda)$, where λ ranges over the set of eigenvalues of A (the poles of system (2)) and φ_λ are a corresponding set of eigenvectors. Said in words, we consider the problem of *eigenstructure identification.*[1]

Our objective is to show that subspace methods provide consistent estimators of the eigenstructure, also for nonstationary cases as above. Of course, none of the matrices $A, B, C, D, K(k)$, and $L(k)$, are known. Matrices $B, D, K(k)$, and $L(k)$, are regarded as nuisance and are not for identification in this paper.

We now introduce the generic subspace algorithm we shall analyze throughout this paper. This generic algorithm will be subsequently specialized to cover the various algorithms used in practice.

[1] This problem and the situation described in (2) naturally occur for example in the modal analysis of mechanical structures subject to vibration under both controlled and/or natural and turbulent excitation [1].

A generic subspace algorithm. Consider an observable pair (C, A) of matrices, where C is $q \times n$ and A is $n \times n$. Throughout this paper, p denotes an integer large enough such that

$$\operatorname{rank}(\mathcal{O}_p) = n, \quad \text{where} \quad \mathcal{O}_p \stackrel{\Delta}{=} \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{p-1} \end{bmatrix} \tag{3}$$

Our generic algorithm assumes a finite family $R_i(N)$ of $q \times r$-matrices, where $r \geq n$, $i = 1, \ldots, p$ and $N > 0$. It returns a pair $(C(N), A(N))$. We describe it next. Consider the matrix $\mathcal{H}_p(N)$ defined by

$$\mathcal{H}_p(N) \stackrel{\Delta}{=} \begin{bmatrix} R_1(N) \\ R_2(N) \\ \vdots \\ R_p(N) \end{bmatrix} \tag{4}$$

and SVD-decompose it as:

$$\begin{aligned} \mathcal{H}_p(N) &= \sum_{i=1}^{\min(pq,r)} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \\ &= \sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T + \sum_{i=n+1}^{\min(pq,r)} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \\ &= \mathbf{U}\ \operatorname{diag}(\sigma_1, \ldots, \sigma_n)\ \mathbf{V}^T + \sum_{i=n+1}^{\min(pq,r)} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \\ &= \mathbf{U}\ S\ \mathbf{V}^T + \sum_{i=n+1}^{\min(pq,r)} \sigma_i \mathbf{u}_i \mathbf{v}_i^T\ . \end{aligned} \tag{5}$$

Partition the $pq \times n$ matrix $\mathbf{U}$ defined in (5) into its p successive q-block rows $\mathbf{U}_1, \ldots, \mathbf{U}_p$ and set

$$\mathbf{U}^{\uparrow} \stackrel{\Delta}{=} \begin{bmatrix} \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_p \end{bmatrix} \quad \text{and} \quad \mathbf{U}^{\downarrow} \stackrel{\Delta}{=} \begin{bmatrix} \mathbf{U}_1 \\ \vdots \\ \mathbf{U}_{p-1} \end{bmatrix}$$

Using these notations, set

$$C(N) \stackrel{\Delta}{=} \mathbf{U}_1 \tag{6}$$

$$A(N) \stackrel{\Delta}{=} \text{least-squares solution of } \mathbf{U}^{\uparrow} = \mathbf{U}^{\downarrow} A \tag{7}$$

Formulas (4–7) constitute our generic subspace algorithm. The remainder of the paper consists in analyzing this algorithm and specializations thereof. The sentence

"$R_i(N)$ provides consistent estimators for (C, A)"

that we use throughout this paper means that, when provided with the sequence $R_i(N)$, this generic algorithm yields consistent estimators $(C(N), A(N))$ for the pair (C, A) in the sense made precise in Theorem 1 below.

3 Basic theorems for nonstationary consistency

Throughout this paper, for t_N a nondecreasing sequence of positive real numbers, $o(t_N)$ generically denotes a matrix-valued sequence M_N, *of fixed dimensions,* such that $M_N/t_N \to 0$ when N tends to infinity.

Also, throughout this paper, we distinguish *Conditions* from *Assumptions.* Assumptions will refer to hypothesized properties of the system or its inputs; Assumptions may or may not hold. In contrast, Conditions can be satisfied by proper design of the algorithm; enforcing these Conditions will be typically part of the process of designing the subspace algorithms.

Our analysis proceeds in two steps. The first step collects the arguments that do not involve probability, whereas only the second step makes use of statistical arguments.

3.1 Non probabilistic analysis

In this subsection, we collect all arguments of the analysis that make no use of probability at all. Therefore, "convergence" is meant here in the usual, non probabilistic, sense.

3.1.1 From Hankel matrices to eigenstructure

For $i = 1, \ldots, p$ and $N > 0$, consider a family $R_i(N)$ of $q \times r$-matrices, satisfying the following condition:

Condition 1 *The matrices $R_i(N), N > 0$, decompose as*

$$R_i(N) \quad = \quad CA^{i-1}G(N) + o(1). \tag{8}$$

Furthermore, the sequence of $n \times r$-matrices $G(N), N > 0$, satisfies the following condition:

$$\liminf_{N\to\infty} \sigma_n\left(G(N)\right) > 0 \ , \tag{9}$$

where $\sigma_n(M)$ denotes the n-th largest singular value of matrix M.

Theorem 1 (consistent estimator [6]) *Under Condition 1, $(C(N), A(N))$ defined by (4–7) is a* consistent estimator *of (C, A) in the following sense:*

> *there exists a sequence of matrices $T(N)$, with $T(N)$ and $T^{-1}(N)$ uniformly bounded w.r.t. N, such that*
>
> $$\lim_{N\to\infty} T^{-1}(N)A(N)T(N) \to A \ , \quad \textit{and} \quad \lim_{N\to\infty} C(N)T(N) \to C \ .$$

Proof: It is found in [6], second part of Section III-C, dealing with the Balanced Realization algorithm. Besides the fact that reference [6] speaks (H, F, G) and not (A, B, C), the only slight change is that matrix $G(N)$ in (8) replaces the controllability matrix $\mathbf{C}(F, G_S)$ of [6], where S is the sample length. ◇

In the following we shall relate our matrices $R_i(N)$ to empirical covariances of data. For this we need some more notations.

3.1.2 Notations

For X and Y two matrices of compatible dimensions, define:

$$\begin{array}{rcl} \langle X, Y\rangle & \stackrel{\Delta}{=} & XY^T \\ \|X\|^2 & \stackrel{\Delta}{=} & \mathrm{Tr}\,\langle X, X\rangle \\ \mathrm{E}(X \mid Y) & \stackrel{\Delta}{=} & \langle X, Y\rangle\langle Y, Y\rangle^{\dagger} Y \\ \mathrm{E}(X \mid Y^{\perp}) & \stackrel{\Delta}{=} & X - \mathrm{E}(X \mid Y) \ , \end{array} \tag{10}$$

where Tr denotes the trace and superscript † denotes the pseudo inverse. For $(y_k)_{k\in\mathbb{Z}}$ a $\mathbb{R}^q$-valued data sequence and $N > 0$ a window length, define

$$Y_i(N) \quad \stackrel{\Delta}{=} \quad \left[\begin{array}{cccc} y_{i+N-1} & \cdots & y_{i+1} & y_i \end{array} \right]$$

and write simply Y_i if no confusion can result. For $(x_k)_{k\in\mathbb{Z}}$ and $(z_k)_{k\in\mathbb{Z}}$ two data sequences of compatible dimensions, we write:

$$\langle X_i, Z_j\rangle_N \triangleq \langle X_i(N), Z_j(N)\rangle \ , \quad \text{and} \quad \mathrm{E}_N(X_i \mid Z_j) \triangleq \mathrm{E}(X_i(N) \mid Z_j(N)) \ .$$

Finally, we shall make use of the following data Hankel matrices:

$$\mathcal{Y}^+_{i,M}(N) \triangleq \begin{bmatrix} Y_{i+M} \\ \vdots \\ Y_{i+2} \\ Y_{i+1} \end{bmatrix}, \ \mathcal{Y}^-_{i,M}(N) \triangleq \begin{bmatrix} Y_i \\ Y_{i-1} \\ \vdots \\ Y_{i-M} \end{bmatrix}, \text{ and } \mathcal{Y}_{i,M}(N) \triangleq \begin{bmatrix} \mathcal{Y}^+_{i,M} \\ \\ \mathcal{Y}^-_{i,M} \end{bmatrix}$$

The above notations are introduced because, depending on the considered algorithms, the data set is indexed as $y_N, \ldots, y_1$ (when only "future" data are needed), or $y_N, \ldots, y_1, y_0, \ldots, y_{-N}$ (when data are split into future and past). Many authors use rather $y_1, \ldots, y_N, y_{N+1}, \ldots, y_{2N}$, or variants thereof. Clearly, the difference is only notational. Also, we have taken identical index M in $\mathcal{Y}^+_{i,M}$ and $\mathcal{Y}^-_{i,M}$ when building $\mathcal{Y}_{i,M}$. Of course, we could take different indices M_+ and M_- without impairing the validity of what follows.

Finally in order to refer to the different algorithms in a systematic way in the sequel, we shall superscript the referred $R_i(N)$ with the index of the corresponding equation. For example,

$$R_i^{(16)}(N) \text{ denotes } R_i(N) \text{ as specified by (16).} \tag{11}$$

3.1.3 Instruments

In this section, we revisit the old concept of "instrument" and use it in our context. Unlike in Section 2 where our problem was stated, we do not distinguish here between observed and unobserved inputs. In the following system, vector ξ collects all inputs of the system considered throughout this section:

$$\left\{ \begin{array}{rcccl} x_k & = & Ax_{k-1} & + & B'\xi_k \\ y_k & = & Cx_{k-1} & + & D'\xi_k \end{array} \right. \tag{12}$$

In (12), $k \in \mathbb{Z}$, x is the $\mathbb{R}^n$-valued state, ξ is the $\mathbb{R}^m$-valued input, and y is the $\mathbb{R}^q$-valued observed output. Fix a window length N. With the notations of Section 3.1.2, system (12) rewrites as follows, for $i = 1, \ldots, p$:

$$\left\{ \begin{array}{rcccl} X_i & = & AX_{i-1} & + & B'\Xi_i \\ Y_i & = & CX_{i-1} & + & D'\Xi_i \end{array} \right. \tag{13}$$

In the following lemma we introduce *instruments* as the key tool in our analysis:

Lemma 1 (instruments) *Let $(z_k)_{k\in\mathbb{Z}}$ be an $\mathbb{R}^M$-valued data sequence and $(s_N)_{N>0}$ an $\mathbb{R}_+$-valued sequence such that*

$$\text{for } j \in \{1,\dots,i\} \ : \ \langle \Xi_j, Z_0\rangle_N = o(s_N) \tag{14}$$

$$\liminf_{N\to\infty} \sigma_n\left(\frac{1}{s_N}\langle X_0, Z_0\rangle_N\right) > 0 \tag{15}$$

Then,

$$R_i(N) \stackrel{\Delta}{=} \frac{1}{s_N}\langle Y_i, Z_0\rangle_N \tag{16}$$

satisfies Condition 1. In the sequel, we call instrument *a signal (z_k) satisfying (14) and (15) for some sequence s_N.*

Proof: The following decompositions hold, for $i > 0$:

$$y_{k+i} \ = \ CA^{i-1}x_k + \sum_{j=1}^{i-1} CA^{i-1-j}B'\xi_{k+j} + D'\xi_{k+i},$$

with the convention that $\sum_1^0 = 0$, and:

$$\begin{aligned} &\sum_{k=0}^{N-1} y_{k+i}z_k^T \\ = \ & CA^{i-1}\sum_{k=0}^{N-1} x_k z_k^T + \sum_{j=1}^{i-1} CA^{i-1-j}\sum_{k=0}^{N-1} B'\xi_{k+j}z_k^T + \sum_{k=0}^{N-1} D'\xi_{k+i}z_k^T \end{aligned} \tag{17}$$

Equation (17) rewrites as follows:

$$\begin{aligned} &\langle Y_i, Z_0\rangle_N \\ = \ & CA^{i-1}\langle X_0, Z_0\rangle_N + \sum_{j=1}^{i-1} CA^{i-1-j}B'\langle \Xi_j, Z_0\rangle_N + D'\langle \Xi_i, Z_0\rangle_N \ , \end{aligned} \tag{18}$$

which proves that $R^{(16)}(N)$ satisfies Condition 1, thanks to (14) and (15). ◇

Lemma 1 and Theorem 1 together ensure that $R^{(16)}(N)$ provides consistent estimators for the pair (C, A)—see (11) for the notational convention used here.

Applying Lemma 1 to system (1) with its combined observed and unobserved inputs can be (tentatively) performed via the following substitutions:

$$\left[\begin{array}{c} B' \\ D' \end{array} \right] \xi_k \;\; = \;\; \left[\begin{array}{c} B \\ D \end{array} \right] u_k + \left[\begin{array}{c} v_k \\ w_k \end{array} \right] \tag{19}$$

Of course, if input ξ_k is observed, *i.e.,* $v_k = w_k = 0$ in (19), then one can chose instrument z_k in such a way that $\langle \Xi_j, Z_0 \rangle_N = 0$ exactly. This is no longer feasible if unobserved inputs exist, since Ξ_j is no longer observed in this case. Therefore, additional work is needed for analyzing system (1) with its combined observed/unobserved inputs. Section 3.2 on probabilistic analysis will address this missing point.

3.1.4 Weighting and Squaring

(This section may be ignored for a first reading.)

As perfectly analyzed in the book [23], there are many different subspace algorithms, and, in addition, each of these possesses a number of variants. Such variants depend on whether the algorithm uses raw data or frequency domain spectra, or time domain covariance matrices as inputs; they also depend on which type of "weighting" is being used. In this section we shall develop a toolbox of lemmas to show that, once one of these variants is shown to be consistent, then so are all related variants. Our toolbox involves the following two tools: *weighting* and *squaring.*

Weighting. Weighting is generally used as part of subspace algorithms and plays an important role in algorithm conditioning and convergence rates. In our case, weighting will be in addition a key tool for the analysis of algorithms, should they be weighted or not.

We distinguish pre-weighting, indicated by the symbol λ in sub- or superscript, and post-weighting, indicated by the symbol ρ in sub- or superscript. Symbols λ and ρ are reminiscent of "left" and "right", respectively. Pre-weighting consists in pre-multiplying the matrix $\mathcal{H}_p$ defined in (4) by a square and invertible matrix W_λ. Post-weighting consists in post-multiplying $\mathcal{H}_p$ by a rectangular matrix W_ρ^T, of dimensions possibly varying with the length N of the record. In this discussion we omit index N when no ambiguity can result. In what follows, superscript ^{w} attached to R_i or $\mathcal{H}_p$ generally announces that weighting will be used in analyzing the corresponding algorithm.

Let r_N be a sequence of positive integers. We are given:

– a family $R_i^w(N)$ of $q \times r_N$-matrices, where $i = 1, \ldots, p$;

– a sequence $W_\lambda(N)$ of pre-weighting matrices of dimensions $pq \times pq$;
– a sequence $W_\rho^T(N)$ of post-weighting matrices of dimensions $r_N \times r$.

Let $\mathcal{H}_p^w(N)$ be the matrix obtained by stacking the matrices $R_i^w(N)$ as in (4). Then, set $\mathcal{H}_p(N) = W_\lambda(N)\,\mathcal{H}_p^w(N)\,W_\rho^T(N)$. Partitioning $\mathcal{H}_p(N)$ as in (4) defines a family $R_i(N)$ of matrices. Now, SVD-decomposing $\mathcal{H}_p(N)$ yields:

$$\mathcal{H}_p(N) \;=\; \mathbf{U}\ \mathrm{diag}(\sigma_1,\ldots,\sigma_n)\ \mathbf{V}^T + \textstyle\sum_{i=n+1}^{\min(pq,r)} \sigma_i \mathbf{u}_i {\mathbf{v}_i}^T \tag{20}$$

For given N, let $(C(N), A(N))$ be the pair obtained by applying formulas (6) and (7) to the matrix $\mathbf{U}$. On the other hand, SVD-decompose $\mathcal{H}_p^w(N)$ as

$$\mathcal{H}_p^w(N) \;=\; \mathbf{U}_w\ \mathrm{diag}(\sigma_1^w,\ldots,\sigma_n^w)\ \mathbf{V}_w^T + \textstyle\sum_{i=n+1}^{\min(pq,r)} \sigma_i^w \mathbf{u}_i^w \mathbf{v}_i^{wT} \tag{21}$$

and set $\mathbf{U}' = W_\lambda \mathbf{U}_w$. Then, let $(C_w(N), A_w(N))$ be the pair obtained by applying formulas (6) and (7) to the matrix $\mathbf{U}'$.

Note that the family $R_i(N)$ possess constant dimensions and is therefore amenable to a direct application of Theorem 1. In contrast, the family $R_i^w(N)$ cannot satisfy Condition 1 since its dimensions are $q \times r_N$ and thus may vary with N. Therefore the consistency of $(C_w(N), A_w(N))$ cannot follow from a direct application of Theorem 1.

Lemma 2 below overcomes this difficulty by making it possible to transfer consistency, from $(C(N), A(N))$ to $(C_w(N), A_w(N))$.

To this end, note that pre- and post-multiplying (21) by $W_\lambda(N)$ and $W_\rho^T(N)$ yields

$$\begin{array}{rcl} \mathcal{H}_p(N) & = & W_\lambda(N)\ \mathbf{U}_w\ \mathrm{diag}(\sigma_1^w,\ldots,\sigma_n^w)\ \mathbf{V}_w^T\ W_\rho^T(N) \\ & + & W_\lambda(N) \left(\textstyle\sum_{i=n+1}^{\min(pq,r)} \sigma_i^w \mathbf{u}_i^w \mathbf{v}_i^{wT} \right) W_\rho^T(N) \end{array} \tag{22}$$

Lemma 2 (weighting) *Assume that the sequence $\mathcal{H}_p(N)$ is bounded w.r.t. N and that the following condition holds:*

$$\limsup_{N\to\infty} W_\lambda(N) \left(\textstyle\sum_{i=n+1}^{\min(pq,r)} \sigma_i^w \mathbf{u}_i^w \mathbf{v}_i^{wT} \right) W_\rho^T(N) \;=\; 0 \tag{23}$$

Then, the pair $(C_w(N), A_w(N))$ is consistent iff the pair $(C(N), A(N))$ is consistent.

Proof: See Appendix A. ◇

Squaring. Squaring is a particular case of post-weighting, where the weighting matrix is just the transpose of the original one. Squaring is an instrumental tool in analyzing projection based algorithms.

Corollary 1 (squaring) *With the same notations as before, assume that* $\mathcal{H}_p(N)$ *and* $\mathcal{H}_p^w(N)$ *are related by* $\mathcal{H}_p(N) = \mathcal{H}_p^w(N)\mathcal{H}_p^w(N)^T$.

1. *If* $\mathcal{H}_p(N)$ *satisfies Condition 1, then the pair* $(C_w(N), A_w(N))$ *is consistent.*
2. *Vice-versa, if* $\mathcal{H}_p^w(N)$ *satisfies Condition 1, then the pair* $(C(N), A(N))$ *is consistent.*

Proof: See Appendix A. ◇

3.2 Probabilistic analysis

So far probabilities were never invoked. In this subsection we collect the arguments involving probability and assumptions of probabilistic nature.

Let us discuss the key conditions allowing us to apply Lemma 1 and Theorem 1 to system (1), taking the unobserved inputs v and w into account.

Suppose first that there is no unobserved input disturbance, *i.e.,* $v = w = 0$ in (1). Then, the ξ_k's introduced in (12) are observed and thus can be explicitly used to satisfy a stronger condition than (14) in Lemma 1, namely $\langle \Xi_j, Z_0 \rangle_N = 0$. Note that no assumption of stochastic nature is required for this reasoning.

Next, consider the opposite case in which there is no observed input, *i.e.,* $B = D = 0$ in (1). Since input disturbances are not observed, the actual values of Ξ_j are unknown when applying Lemma 1 and therefore cannot be used while constructing the instrument z_k.

This problem, however, can be solved by using *stochastic* knowledge about unobserved input disturbances. To this end, we now introduce the needed probabilistic setting, and, prior to this, the martingale argument we shall use.

3.2.1 A martingale argument

Lemma 3 *Let* $(v_k)_{k\geq 0}$ *and* $(z_k)_{k\geq 0}$ *be two sequences of square integrable vector valued random variables defined over some probability space* $(\Omega, \mathcal{G}, \mathbb{P})$ *and let* $(\mathcal{G}_k)_{k\geq 0}$ *be an increasing family of sub-*σ*-algebras of* $\mathcal{G}$ *such that:*

$$\begin{array}{r} \sup_{k\geq 0} \mathbb{E}\|v_k\|^2 \leq K < \infty \ , \ \textit{and} \ \lim_{N\to\infty} \sum_{k=0}^{N} \|z_k\|^2 = +\infty \ \textit{w.p.1} \ ; \\ v_k \ \textit{and} \ z_k \ \textit{are} \ \mathcal{G}_k\textit{-measurable, and} \ \mathbb{E}(v_k \mid \mathcal{G}_{k-1}) = 0 \ . \end{array} \tag{24}$$

Then, for any $j > 0$, the following holds:

$$\lim_{N\to\infty} \frac{M_N(j)}{\sum_{k=0}^{N} \|z_k\|^2} = 0 \ \textit{w.p. 1}, \quad \textit{where} \quad M_N(j) \stackrel{\Delta}{=} \sum_{k=j}^{N} v_k z_{k-j}^T \,. \tag{25}$$

Nota In formula (24), the conditional expectation $\mathbb{E}(. \mid \mathcal{G}_{k-1})$ should not be confused with our matrix projection operator $\mathrm{E}(. \mid .)$ in (10).

Proof: It is a mild variation of the argument of [6], Section III-A. We repeat it here for the sake of completeness. Since we can reason on each entry of matrix M_N separately, we can, without loss of generality, assume that v_k and z_k are both scalar signals. By the second condition of (24), we know that $(M_k)_{k\geq 0}$ is a square integrable scalar martingale w.r.t. $(\mathcal{G}_k)_{k\geq 0}$. By (24), we have $\mathbb{E}((M_k - M_{k-1})^2 \mid \mathcal{G}_{k-1}) = \mathbb{E}(v_k^2 \mid \mathcal{G}_{k-1}) z_{k-j}^2 = \mathbb{E}(v_k^2) z_{k-j}^2 \leq K z_{k-j}^2$. The proof is then completed by using Theorem 2 below, which can be found in [15, 20]. ⋄

The real-valued stochastic process $(M_k)_{k\geq 0}$ is called a *locally square integrable martingale* w.r.t. $(\mathcal{G}_k)_{k\geq 0}$ if $1/\ \mathbb{E}(M_l \mid \mathcal{G}_{l-1}) = 0$, and $2/\ \forall L < \infty$, $\sup_{l\leq L} \mathbb{E} M_l^2 < \infty$.

Theorem 2 ([15, 20]) *Let $(M_k)_{k\geq 0}$ be a locally square integrable martingale w.r.t. $(\mathcal{G}_k)_{k\geq 0}$, such that $M_0 = 0$. Set*

$$[M, M]_k \quad = \quad \sum_{l=1}^{k} \mathbb{E}((M_l - M_{l-1})^2 \mid \mathcal{G}_{l-1}) \ .$$

*Then, the following two properties hold w.p.*1*:*

$$\frac{M_k}{[M, M]_k} \to 0 \ \textit{on the set} \ \{\lim_{k\to\infty} [M, M]_k = +\infty\} \ ,$$
$$\lim_{k\to\infty} M_k \ \textit{exists and is finite on the set} \ \{\lim_{k\to\infty} [M, M]_k < +\infty\}.$$

3.2.2 Analyzing the generic subspace algorithm

In this section we combine the results from Sections 3.1.3 and 3.2.1 to handle system (1) with its combined observed/unobserved inputs. We repeat again system (1) for convenience:

$$\left\{ \begin{array}{rcccccc} x_k & = & Ax_{k-1} & + & Bu_k & + & v_k \\ y_k & = & Cx_{k-1} & + & Du_k & + & w_k \end{array} \right. \tag{26}$$

where $k \in \mathbb{Z}$, y is the $\mathbb{R}^q$-valued observed output, x is the $\mathbb{R}^n$-valued state, u is the $\mathbb{R}^m$-valued observed input, (v, w) is an unobserved input disturbance.

To be able to use stochastic information on the unobserved inputs v, w we assume that all variables arising in system (26) are defined over some probability space $(\Omega, \mathcal{F}, \mathbb{P})$.

Available information is captured by the following σ-algebras:

$$\begin{aligned} \mathcal{F}_k &\stackrel{\Delta}{=} \underbrace{\sigma\left(u_j : j \in \mathbb{Z}\right)}_{\mathcal{F}^u} \vee \underbrace{\sigma\left(y_l, v_l, w_l : l \leq k\right)}_{\mathcal{F}_k^{y,v,w}} \\ \mathcal{F}_k^o &\stackrel{\Delta}{=} \underbrace{\sigma\left(u_j : j \in \mathbb{Z}\right)}_{\mathcal{F}^u} \vee \underbrace{\sigma\left(y_l : l \leq k\right)}_{\mathcal{F}_k^y} \end{aligned}$$

σ-algebra $\mathcal{F}^u$ is the information provided by the entire observed input sample; σ-algebra $\mathcal{F}_k^{y,v,w}$ is the information provided by the unobserved inputs v and w and the output y up to time k; finally, σ-algebra $\mathcal{F}_k^y$ is the information provided by the only output y up to time k. Regarding the unobserved inputs, we assume the following:

Assumption 1 (unobserved inputs) *Stochastic inputs v and w satisfy the following conditions:*

$$\sup_{k \geq 0} \mathbb{E}\left(\|v_k\|^2 + \|w_k\|^2\right) < \infty,$$
$$\forall j > 0, \forall k \geq 0 \ : \ \mathbb{E}\left(v_{k+j} \mid \mathcal{F}_k\right) = 0 \ \textit{and} \ \mathbb{E}\left(w_{k+j} \mid \mathcal{F}_k\right) = 0 \ .$$

Note that these conditions do not request any kind of stationarity. Assumption 1 involves the joint distribution of v_k, w_k, and u_k. It is in particular satisfied when observed and unobserved inputs are independent. Besides Assumption 1, no condition is required on the statistics of the observed input u_k. Consider the following conditions regarding instruments:

Condition 2 (instruments) *Instrument (z_k) satisfies the following conditions:*

$$z_k \ \textit{is} \ \mathcal{F}_k^o\textit{-measurable} \tag{27}$$

$$\lim_{N \to \infty} s_N = \infty, \ \textit{where} \ s_N \stackrel{\Delta}{=} \sum_{k=-M}^{N-1} \|z_k\|^2 \tag{28}$$

$$\left\langle \begin{bmatrix} B \\ D \end{bmatrix} U_j, Z_0 \right\rangle_N = o(s_N) \ \textit{for} \ j > 0 \tag{29}$$

$$\liminf_{N \to \infty} \sigma_n \left(\frac{1}{s_N} \langle X_0, Z_0 \rangle_N \right) > 0 \tag{30}$$

Property (27) guarantees that instrument z_k depends only on observed quantities. Integer $M \geq 0$ in (28) is a constant selected according to each particular instance of the family $R_i(N)$. Property (28) expresses that instrument (z_k) possesses sustained energy.

Covariance based subspace. The following theorem is our first main result. It provides the analysis of algorithms of the form (16), *i.e.,* covariance based ones.

Theorem 3 (covariance based subspace) *Assume that Assumption 1 regarding unobserved inputs, and Condition 2 regarding instruments, are in force. Then,* $R_i^{(16)}(N)$ *satisfies Condition 1, with probability* 1*.*

In other words, the set of trajectories of the system for which Condition 1 is satisfied has probability 1. Pick any trajectory in this set, we can apply Theorem 1, which shows that, for this trajectory, our generic algorithm provides consistent estimators in the sense of Theorem 1. This shows that our generic algorithm provides consistent estimators in the statistical sense (convergence w.p.1 to the true value for the parameters to be estimated).

Proof: Using the notations of Section 3.1.2, system (26) writes as follows, for $i = 1, \ldots, p$:

$$\left\{ \begin{array}{rcl} X_i & = & AX_{i-1} + BU_i + V_i \\ Y_i & = & CX_{i-1} + DU_i + W_i \end{array} \right. \tag{31}$$

On the other hand, system (26) yields the following decomposition for $y_{k+i}, i > 0$ (we use the convention that $\sum_1^0 = 0$):

$$y_{k+i} = CA^{i-1}x_k + \sum_{j=1}^{i-1} CA^{i-1-j}\widehat{v}_{k+j} + \widehat{w}_{k+i}$$

where $\widehat{v}_k \stackrel{\Delta}{=} Bu_k + v_k$ and $\widehat{w}_k \stackrel{\Delta}{=} Du_k + w_k$. Using the notations of Section 3.1.3, this decomposition rewrites as follows, for $i > 0$:

$$Y_i = CA^{i-1}X_0 + \sum_{j=1}^{i-1} CA^{i-1-j}\widehat{V}_j + \widehat{W}_i \tag{32}$$

where $\widehat{V_i} \stackrel{\Delta}{=} BU_i + V_i$ and $\widehat{W_i} \stackrel{\Delta}{=} DU_i + W_i$. Note that

$$\langle V_j, Z_0 \rangle_N \quad = \quad \sum_{k=0}^{N-1} v_{k+j} z_k^T, \tag{33}$$

and a similar formula holds with W_j instead of V_j. By (27) and (28) of Condition 2, instrument (z_k) satisfies (24) in Lemma 3. By Assumption 1, noises v_k and w_k satisfy (24) in Lemma 3, with $\mathcal{F}_k$ substituted for $\mathcal{G}_k$. Therefore Lemma 3 can be applied with $\mathcal{F}_k$ substituted for $\mathcal{G}_k$, which yields, with probability 1:

$$\forall j \in \{1, \ldots, p\} \quad : \quad \left\langle \begin{bmatrix} V_j \\ W_j \end{bmatrix}, Z_0 \right\rangle_N = o(s_N) \tag{34}$$

Set

$$\begin{aligned} \xi_k \quad &\stackrel{\Delta}{=} \quad \begin{bmatrix} B \\ D \end{bmatrix} u_k + \begin{bmatrix} v_k \\ w_k \end{bmatrix} \\ B' \quad &\stackrel{\Delta}{=} \quad [\ I_n \quad 0_q\ ] \\ D' \quad &\stackrel{\Delta}{=} \quad [\ 0_n \quad I_q\ ] \end{aligned}$$

where the subscripts $_n$ and $_q$ indicate the dimensions of the corresponding matrices. Using this change of notation allows us to rewrite system (26) in the form (12) used in Lemma 1.

Consider now Condition 2. Combining (34) with (29) shows that system (12) satisfies (14) in Lemma 1. On the other hand (15) in Lemma 1 is ensured by Property (30) of Condition 2. Therefore, by Lemma 1 we conclude that Condition 1 is satisfied, with probability 1. ◇

Remark. In fact our method could accomodate as well additional "small" perturbations in system (26), *i.e.,* additional inputs μ_k and ν_k in state and observation equations respectively, such that

$$\frac{1}{s_N} \sum_{k=-M}^{N-1} \|\mu_k\|^2 + \|\nu_k\|^2 \quad = \quad o(1)\ .$$

Transient terms or leakage effects such as considered in [8, 9] are covered by these additional terms, and therefore do not impair nonstationary consistency.

Projection based subspace. Projection based subspace methods, *i.e.,* methods of the form

$$R_i(N) \stackrel{\Delta}{=} \frac{1}{s_N} \mathrm{E}_N(Y_i \mid Z_0) \tag{35}$$

are in fact more popular than covariance based ones, see [23]. They are often referred to as "data based" subspace methods. Unfortunately, these methods cannot be handled directly by Theorem 3. In fact, Theorem 1 itself does not apply. The reason for this is simple: $R_i^{(35)}(N)$ has dimensions $q \times dim(Z_0(N))$. So its dimensions vary with N and therefore Theorem 1 cannot apply. Fortunately, the weighting technique of Section 3.1.4 can be used to overcome this difficulty as we shall see now.

Corollary 1 of Section 3.1.4 can be used to relate covariance based methods, *i.e.,* of the form:

$$R_i'(N) \quad = \quad \frac{1}{s_N} \langle Y_i, Z_0 \rangle_N$$

to projections based ones:

$$R_i(N) \quad = \quad \frac{1}{s_N} \mathrm{E}_N(Y_i \mid Z_0). \tag{36}$$

The former are handled by Theorem 3 but the latter are not. To establish this relation, stack the matrices $R_i(N)$ as usual now and consider

$$\mathcal{H}_p(N) = \mathrm{E}_N(\mathcal{Y}_{0,n}^+ \mid Z_0). \tag{37}$$

Note that this Hankel matrix has its second dimension that varies with the length N of the data sample and thus cannot be handled by Theorem 3. To study this algorithm, we shall therefore use a "squaring" method based on Corollary 1. Consider

$$\begin{aligned} \mathcal{K}_p(N) &= \mathcal{H}_p(N)\, \mathcal{H}_p(N)^T \\ &= \langle \mathcal{Y}_{0,p}^+, Z_0 \rangle_N \, \langle Z_0, Z_0 \rangle_N^\dagger \, \langle Z_0, \mathcal{Y}_{0,p}^+ \rangle_N \end{aligned}$$

By point 1 of Corollary 1 it is enough to guarantee that $\mathcal{K}_p(N)$ satisfies Condition 1. To this end, renormalize instrument Z_0:

$$\widehat{Z}_0 \quad = \quad \langle Z_0, Z_0 \rangle_N^{-\frac{1}{2}} Z_0(N), \tag{38}$$

where superscript $^{-1/2}$ denotes the square root of the pseudo-inverse. Note that (38) amounts to whitening instrument z_k. Then, $\mathcal{K}_p(N)$ rewrites:

$$\mathcal{K}_p(N) \quad = \quad \langle \mathcal{Y}_{0,p}^+, \widehat{Z}_0 \rangle_N \, \langle \widehat{Z}_0, \mathcal{Y}_{0,p}^+ \rangle_N \tag{39}$$

By point 2 of Corollary 1 it is enough to guarantee that the square root matrix $\langle \mathcal{Y}_{0,p}^{+}, \widehat{Z}_0 \rangle_N$ satisfies Condition 1. Note that instrument $\widehat{Z}_0$ satisfies the measurability property (27) of Condition 2. Our second main result therefore states as follows:

Theorem 4 (projection based subspace) *Assume that Assumption 1 is in force, as well as properties (28–30) of Condition 2, for instrument (38). Then,* $R_i^{(36)}(N)$ *satisfies Condition 1, with probability* 1.

Consequently, $R_i^{(36)}(N)$ yields a consistent subspace algorithm, in the statistical sense.

3.3 Discussing Assumptions and Conditions

Here we collect remarks concerning our Assumptions and Conditions.

What if matrix A is unstable? Strictly speaking, it is nowhere required that matrix A shall be stable. However, if A is has some unstable eigenvalues and some stable ones, then property (30) of Condition 2 can hardly be satisfied.

What can the observed inputs u be? Property (29) of Condition 2 relates instrument z_k to input u_k; but the latter condition should rather be seen as a condition on the instrument, not as a condition on the input. The only important requirement on u is Assumption 1, since it requires that future unobserved inputs $v_{k+j}, w_{k+j}, j > 0$ shall be independent from past inputs $u_{k-l}, l \geq 0$. For example, if (v, w) is white noise, then this prevents u_k from depending on future outputs $y_{k+j}, j > 0$. On the other hand, there is no requirement per se that u should be stationary. In some sense, the probability distribution of u does not matter and we regard u as stochastic in Section 3.2.2 only for mathematical convenience.

What is really allowed regarding unobserved input noise (v, w)*? Can it be colored?* First, the time-varying matrices $K(k)$ and $L(k)$ in (2) may be random. This must, however, occur in a way that Assumption 1 shall not be invalidated. For example, referring to (2), it is possible that $K(k)$ and $L(k)$ are stochastic processes that are independent from both underlying white noise ν_k and observed input u_k.

Can (v, w) be colored in (1)? Yes in part. In fact, moving average measurement noise is allowed:

$$w_k = \sum_{j=0}^{J} L_j \nu_{k-j},$$

where ν is a possibly nonstationary white noise. To see this, rewrite (1) as follows, with $\xi_k^T \stackrel{\Delta}{=} [\nu_k^T, \ldots, \nu_{k-J+1}^T]$:

$$\left\{ \begin{array}{rcccccccc} x_k & = & & Ax_{k-1} & & + & Bu_k & + & v_k \\ \xi_k & = & & S\xi_{k-1} & & + & & + & T\nu_k \\ y_k & = & \left[\begin{array}{cc} C & \overline{L} \end{array} \right] & \left[\begin{array}{c} x_{k-1} \\ \xi_{k-1} \end{array} \right] & & + & Du_k & + & L_0\nu_k \end{array} \right. \tag{40}$$

where S is the nilpotent matrix having 1's on the lower diagonal entries and 0's elsewhere, $T = [I\, 0 \ldots 0]^T$, and $\overline{L} = [L_1 \ldots L_J]$. Applying the generic algorithm with (16) to system (40) yields the desired eigenstructure of pair (C, A), provided that we know that A does not have 0 as an eigenvalue.

4 Analysis of some subspace algorithms

In this section we apply our toolbox of theorems and lemmas to sample subspace methods. To avoid boring notational adjustments, we keep our notational conventions and will therefore sometimes deviate from the original presentations in this respect.

Key conditions ensuring nonstationary consistency are Assumption 1 and Condition 2. Assumption 1 involves the unobserved inputs, we assume it to hold throughout this section and will not discuss it any further. In contrast, Condition 2 is a *design constraint on the selection of the instruments*: this is the key condition to be verified or enforced when analyzing specific algorithms.

Regarding the details of Condition 2, we shall pay great attention to verifying that (27) and (29) are satisfied, as these conditions drive the choice of the instruments. Condition (30) amounts to requiring that the instrument is well correlated to the state. In contrast, we shall not discuss the satisfaction of condition (28); this condition just translates, for each particular algorithm, into corresponding conditions for the original system (26).

Finally, checking for consistency requires that proper normalization is applied. This is the very role of the scaling factor s_N. In practice the algorithms are applied with given sample length N, and then, scaling is just an irrelevant issue. Therefore, we shall ignore scaling in this section.

4.1 Output-only (OO) subspace algorithms

By definition, these algorithms assume $B = D = 0$ in (26). Therefore (29) in Condition 2 is trivially satisfied, thus we essentially need to check the measurability property (27).

Basic OO subspace algorithm This is the simplest algorithm to analyze. Introduce the instrument

$$z_k \stackrel{\Delta}{=} \begin{bmatrix} y_k \\ \vdots \\ y_{k-M} \end{bmatrix} \tag{41}$$

and take

$$R_i(N) = \langle Y_i, Z_0 \rangle_N. \tag{42}$$

Instrument (41) satisfies (27) in Condition 2. Hence Theorem 3 applies and proves consistency of $R_i^{(42)}(N)$. Note that

$$\langle X_0, Z_0 \rangle_N = \left[\begin{array}{cccc} F(N) & AF(N) & \dots & A^M F(N) \end{array} \right],$$

where $F(N) = \langle X_0, Y_0 \rangle_N$. Hence, (30) can be interpreted as y_k being "uniformly of order n".

Covariance driven OO subspace algorithm [6, 21, 1, 14] This algorithm is a variation of the previous algorithm, it was however proposed earlier. It consists in computing, for $i = 1, \dots, p$:

$$\begin{array}{l} R_i(N) = \left[\begin{array}{cccc} \hat{r}_i(N) & \hat{r}_{i+1}(N) & \cdots & \hat{r}_{i+M}(N) \end{array} \right] \\ \text{where } \hat{r}_j(N) = \langle Y_j, Y_0 \rangle_N \end{array} \tag{43}$$

With instrument z_k as in (41), we have

$$R_i^{(43)}(N) - \langle Y_i, Z_0 \rangle_N = \left[\begin{array}{ccc} \delta\langle Y_i, Y_0 \rangle & \dots & \delta\langle Y_i, Y_{-M} \rangle \end{array} \right]$$

where $\delta\langle Y_i, Y_{-k} \rangle \stackrel{\Delta}{=} \langle Y_i, Y_{-k} \rangle_N - \langle Y_{i+k}, Y_0 \rangle_N$ is such that

$$\| \delta\langle Y_i, Y_{-k} \rangle \| \leq 2\, s^*_{M,N},$$

where

$$s^*_{M,N} \stackrel{\Delta}{=} \sup_{-M\leq j\leq N-M} \sum_{l=j}^{j+M} \|y_l\|^2 \ .$$

This implies

$$\|R_i^{(43)}(N) - \langle Y_i, Z_0\rangle_N\| \quad = \quad o(s_N) \ , \tag{44}$$

provided that the following assumption holds:

Assumption 2 *For M fixed, $s^*_{M,N} = o(s_N)$.*

Under the above additional assumption, (44) holds and therefore, instrument z_k defined in (41) satisfies Condition 2. Therefore, by Theorem 3, we derive that $R_i^{(43)}(N)$ yields a consistent subspace algorithm.

Data driven OO subspace algorithms [23]. This algorithm is found in [23]– Theorem 8, Chapter 3. It consists in computing

$$\mathcal{H}_p(N) = \mathrm{E}_N(\mathcal{Y}^+_{0,p} \mid \mathcal{Y}^-_{0,M}). \tag{45}$$

To study this algorithm, we shall use Theorem 4 about projection based methods. To this end, set

$$\widehat{Z}_0 \quad = \quad \left(\langle \mathcal{Y}^-_{0,M}, \mathcal{Y}^-_{0,M}\rangle_N\right)^{-\frac{1}{2}} \mathcal{Y}^-_{0,M}(N), \tag{46}$$

which amounts to whitening the instrument (41). Instrument (46) satisfies the measurability condition (27) of Condition 2. Assuming that (28) and (30) are satisfied, this yields consistency, by Theorem 4.

4.2 Input-output (IO) subspace algorithms

Many variants have been considered. We review some representative ones.

Covariance driven IO subspace algorithms with projection on the orthogonal of the input [14] This algorithm consists in computing (cf. notations (10)):

$$\hat{r}_i(N) \quad = \quad \langle Z_i, Z_0\rangle_N, \text{ where } Z_i \stackrel{\Delta}{=} \mathrm{E}_N\left(Y_i \,\middle|\, \mathcal{U}^{\perp}_{0,M}\right) \tag{47}$$

$$R_i(N) \quad = \quad \left[\ \hat{r}_i(N) \quad \hat{r}_{i+1}(N) \quad \dots \quad \hat{r}_{i+M}(N)\ \right] \tag{48}$$

First, note that $\langle Z_i, Z_0\rangle_N = \langle Y_i, Z_0\rangle_N$. The associated instrument z_k is therefore the sequence of the successive columns of matrix Z_0. Note that z_k is $\mathcal{F}_k^o$-measurable. The rest of the analysis of this algorithm proceeds as for $R_i^{(43)}(N)$. Property (30) can be seen as that z_k itself being "uniformly of order n".

Data driven IO subspace algorithm with projection on the orthogonal of the input [23]. This algorithm is known as the Projection algorithm in [23] – Chapter 2.3.2. It consists in computing $\mathcal{H}_p = \mathrm{E}_N(\mathcal{Y}_{0,p}^+ \mid \mathcal{Z}_{0,M}^-)$, where Z_i is as in (47). We conclude as for $R_i^{(45)}(N)$.

Data driven subspace algorithm using projected inputs as instruments [25, 26, 10]. This algorithm was first proposed in [25, 26] under the name of PI-MOESP. It was studied recently in [10, 11]; a detailed presentation is found in [11]. It consists in computing a (left) weighted version of

$$\mathcal{H}_p \;=\; \mathrm{E}_N(\mathcal{Y}_{0,p}^+ \mid \mathcal{L}_{0,M}^-), \tag{49}$$

where $\mathcal{L}_{0,M}^-$ is obtained, with the notations of Section 3.1.2, by stacking

$$L_i \;\stackrel{\Delta}{=}\; \mathrm{E}_N\left(U_i \,\middle|\, \left(\mathcal{U}_{0,M}^+\right)^{\perp}\right), \text{ for } i = -M, \ldots, 0. \tag{50}$$

Introduce the following instrument:

$$Z_0 \;=\; \left(\langle \mathcal{L}_{0,M}^-, \mathcal{L}_{0,M}^-\rangle_N\right)^{-\frac{1}{2}} \mathcal{L}_{0,M}^-(N). \tag{51}$$

The squaring argument already used in analyzing (45) can be used here too. Once more, instrument (51) satisfies the measurability property (27) in Condition 2, and we conclude as for (45). Note that, to get this measurability condition, it was essential that the observation σ-algebra $\mathcal{F}_k^o$ contains both past and future of the observed input u.

Note also that we could have equally well used Y_i, or $\begin{bmatrix} U_i \\ Y_i \end{bmatrix}$, instead of U_i in (50), thus obtaining two variants of the above method.

Data driven subspace identification using oblique projections [22, 23]. This category includes popular subspace algorithms such as N4SID and MOESP [22,

23] as well as any variation of them by using weights, including the CVA method [23]. We focus on N4SID and MOESP.

The popular N4SID algorithm of [22] and [23] – Section 4.3.1, consists in computing the so-called *oblique projection* of $\mathcal{Y}_{0,p}^+$ on $\begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix}$ along $\mathcal{U}_{0,M}^+$,

$$\mathcal{H}_p(N) \;=\; \mathcal{Y}_{0,p}^+ \Big/_{\mathcal{U}_{0,M}^+} \begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix}, \tag{52}$$

which is shown to rewrite as follows (see formula (1.7) of [23]):

$$\mathcal{H}_p(N) \;=\; \mathcal{H}_p^w(N) \left[\mathrm{E}_N \left(\begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix} \,\middle|\, \left(\mathcal{U}_{0,M}^+ \right)^\perp \right) \right]^\dagger \begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix}, \tag{53}$$

where

$$\mathcal{H}_p^w(N) = \mathrm{E}_N \left(\mathcal{Y}_{0,p}^+ \,\middle|\, \left(\mathcal{U}_{0,M}^+ \right)^\perp \right) .$$

Define

$$\mathcal{H}_p'(N) \;\stackrel{\Delta}{=}\; \mathcal{H}_p^w(N) W_\rho^T(N), \tag{54}$$

where

$$W_\rho^T(N) = \left[\mathrm{E}_N \left(\begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix} \,\middle|\, \left(\mathcal{U}_{0,M}^+ \right)^\perp \right) \right]^\dagger \left(\begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix} \begin{bmatrix} \mathcal{U}_{0,M}^- \\ \mathcal{Y}_{0,M}^- \end{bmatrix}^T \right)^{1/2}$$

(Note that the expression for $\mathcal{H}_p'(N)$ corresponds to the formulation of N4SID given in [28].) Set $\mathcal{K}(N) = \mathcal{H}_p(N)\mathcal{H}_p(N)^T$ and $\mathcal{K}'(N) = \mathcal{H}_p'(N)\mathcal{H}_p'(N)^T$. Note that $\mathcal{K}(N) = \mathcal{K}'(N)$. By using repeatedly the same squaring argument as in the proof of Theorem 4, we deduce that, if $\mathcal{H}_p'(N)$ satisfies Condition 1, then so does $\mathcal{H}_p(N)$.

Now, assume for a while that we are able to prove that $\mathcal{H}_p^w(N)$ satisfies Condition 1. Then, according Lemma 2, the pair of matrices $(C(N), A(N))$ corresponding to $\mathcal{H}_p'(N)$ will be consistent provided that (23) holds, with $W_\lambda(N) = I$.

To prove that $\mathcal{H}_p^w(N)$ satisfies Condition 1, note that $\mathcal{H}_p^w(N)$ rewrites

$$\mathcal{H}_p^w(N) \;=\; \mathrm{E}_N(\mathcal{Y}_{0,p}^+ \mid Z_0)$$

where Z_0 is a basis for the orthogonal complement of $\mathcal{U}^+_{0,M}$ in the space generated by

$$\left[\begin{array}{c} \mathcal{U}^+_{0,M} \\ \mathcal{U}^-_{0,M} \\ \mathcal{Y}^-_{0,M} \end{array} \right]$$

$\mathcal{H}^w_p(N)$ can therefore be analyses in the very same way as (49).

The following remark can be stated about (30) and (23). These conditions are fragile if $W^T_\rho(N)$ in (54) is close to having rank less than p, which happens when the future $\mathcal{U}^+_{0,M}$ of input u is almost parallel to

$$\left[\begin{array}{c} \mathcal{U}^-_{0,M} \\ \mathcal{Y}^-_{0,M} \end{array} \right]$$

The latter fact is indeed known from the practice about N4SID and is also analyses in [12].

The same analysis also applies to the MOESP algorithm described in [23] – Section 4.3.2, [25, 26], and [3]. This algorithm consists in computing

$$R_i(N) \;\; = \;\; R^{(52)}_i(N)\, \Pi_{\left(\mathcal{U}^+_{0,M}\right)^\perp} \;,$$

where $\Pi_{\left(\mathcal{U}^+_{0,M}\right)^\perp}$ denotes the (orthogonal) projection on $\left(\mathcal{U}^+_{0,M}\right)^\perp$. Thus, MOESP amounts to computing

$$\mathcal{H}_p(N) = \mathcal{H}^w_p(N) W^T_\rho(N)^\dagger W^T_\rho(N), \tag{55}$$

where

$$W^T_\rho(N) \;\; \stackrel{\Delta}{=} \;\; \mathrm{E}_N\left(\left[\begin{array}{c} \mathcal{U}^-_{0,M} \\ \mathcal{Y}^-_{0,M} \end{array} \right] \middle| \left(\mathcal{U}^+_{0,M}\right)^\perp \right) \;. \tag{56}$$

Following the same lines as for N4SID, MOESP yields consistent estimates. The same remark as for N4SID applies, regarding the conditioning of $W^T_\rho(N)$ in (56) and its impact on the behavior of the algorithm. The class of subspace methods described in [23], including CVA, is analyses along the same lines.

Covariance driven subspace algorithm using projected past inputs and outputs as instruments [28] Those methods encompass the methods also known as IVM, CVA, PO-MOESP and N4SID in their covariance form [28]. In this paper, we will focus on the unweighted IV related to $\mathcal{H}_p$ defined as

$$\mathcal{H}_p \;=\; \langle \mathcal{Y}^+_{0,p}, \mathcal{L}^-_{0,M} \rangle_N \;, \tag{57}$$

where $\mathcal{L}^-_{0,M}$ is defined by stacking for $i = -M, \ldots, 0$

$$L_i \;\triangleq\; \mathrm{E}_N\left(\left[\begin{array}{c} U_i \\ Y_i \end{array} \right] \middle| \left(\mathcal{U}^+_{0,M} \right)^\perp \right) \;. \tag{58}$$

The rest of the analysis of this algorithm proceeds as for (48).

4.3 Time– vs. frequency–domain.

For $(y_k)_{k\in\mathbb{Z}}$ an $\mathbb{R}^q$-valued data sequence and $N > 0$ a window length, the DFT of $Y_i(N)$, denoted by $\widehat{Y}_i(N)$, is equal to

$$\widehat{Y}_i(N) \;=\; Y_i(N)\, \Delta^q_N \;, \tag{59}$$

where (in (60), $\otimes$ denotes the Kronecker product):

$$\Delta^q_N \triangleq \frac{1}{N^{q/2}} \left[\begin{array}{ccc} e^{-2j\pi \frac{0}{N}} & \ldots & e^{-2j\pi \frac{0N}{N}} \\ e^{-2j\pi \frac{1}{N}} & \ldots & e^{-2j\pi \frac{1N}{N}} \\ \vdots & \vdots & \vdots \\ e^{-2j\pi \frac{N-1}{N}} & \ldots & e^{-2j\pi \frac{(N-1)N}{N}} \end{array} \right] \otimes I_q \tag{60}$$

Since matrix Δ^q_N is orthogonal, then

$$\langle \widehat{X}, \widehat{Y} \rangle_N = \langle X, Y \rangle_N,$$

and

$$\mathrm{E}_N(\widehat{X} \mid \widehat{Y}) = \mathrm{E}_N(X \mid Y)\, \Delta^q_N \;.$$

Hence, Condition 2 can be considered equivalently in the time domain or in the frequency domain. Therefore, frequency domain subspace algorithms corresponding to [17, 18] behave exactly the same way as their time domain counterparts regarding nonstationary consistency.

5 Conclusion

We have revisited eigenstructure identification via subspace methods. This problem is clearly easier than full system matrix identification. On the other hand, consistency of eigenstructure identification still holds for nonstationary inputs (in fact, for "nonstationary zero part").

For this study, we have adapted the original method of [6]. We believe that our presentation enlightens the reasons for subspace methods to converge, and therefore can serve as a guideline for further new designs. Our analysis shows that the old fashioned "instruments" are still a useful concept in this respect.

Martingale techniques were used to deal with unobserved inputs—for unobserved inputs, "deterministic" projections based on observed data cannot be used; they can be replaced by "stochastic" projections via conditional expectations. This technique requires a probabilistic setting for the unobserved inputs, and the white noise assumption provides a situation in which finding instruments is easy. This suggests that our martingale approach could possibly be replaced by any other method providing orthogonality conditions without the need for observing data.

Not surprisingly, transient and leakage effects are not an issue for nonstationary consistency. And the results equivalently apply to both time- and frequency-domain methods.

Finally, we have only studied nonstationary consistency, not nonstationary convergence rates. The latter subject is definitely much harder. The only results we are aware of in this direction are found in [19].

Acknowledgement. *Michèle Basseville is gratefully acknowledged for corrections and useful suggestions on an earlier version of the manuscript. Also, the reviewers have significantly helped improving the revised version.*

A Missing proofs of Section 3.1.4

Proof of Lemma 2. The proof relies on the Lemma 2 in Appendix C of [6], which we repeat here for completeness:

Lemma 4 ([6]) *Let $\mathcal{K}(N)$ and $\mathcal{K}'(N)$ be two sequences of matrices of fixed dimensions, satisfying the following conditions:*

(i) The sequence $\mathcal{K}(N)$ is bounded and $\mathcal{K}(N) - \mathcal{K}'(N) \to 0$ when $N \to \infty$.

(ii) For every N, the SVD of matrix $\mathcal{K}(N)$ is $\mathcal{K}(N) = \mathbf{U}\ \mathrm{diag}(\sigma_1, \ldots, \sigma_n)\, \mathbf{V}^T$, and $\liminf_{N\to\infty} \sigma_n > 0$ *holds.*

SVD-decompose $\mathcal{K}'(N)$ as $\mathcal{K}'(N) = \mathbf{U}'\ \mathrm{diag}(\sigma'_1, \ldots, \sigma'_n)\, \mathbf{V'}^T +$ higher order terms. *Then, there exists a sequence of $p \times p$-matrices $T(N)$, bounded with bounded inverse, such that $\mathbf{U}'(N) - \mathbf{U}(N)T(N) \to 0$ when $N \to \infty$.*

Return to the proof of Lemma 2. Set $\mathcal{K}'(N) = \mathcal{H}_p(N)$ and

$$\mathcal{K}(N) \quad = \quad W_\lambda(N)\ \mathbf{U}_w\ \mathrm{diag}(\sigma_1^w, \ldots, \sigma_n^w)\ \mathbf{V}_w^T\, W_\rho^T(N).$$

By (22) and (23), we have $\mathcal{K}(N) - \mathcal{K}'(N) \to 0$. On the other hand, it is assumed for Lemma 2 that $\mathcal{K}'(N) = \mathcal{H}_p(N)$ is bounded. Therefore, Lemma 4 applies. Since $\mathcal{K}(N)$ has rank exactly p, the left most factor in the SVD of $\mathcal{K}(N)$ is obtained from $W_\lambda(N)\mathbf{U}_w$ by a post-multiplication by an invertible matrix. On the other hand, the left factor $\mathbf{U}'$ associated to $\mathcal{K}'(N) = \mathcal{H}_p(N)$ by Lemma 4 coincides with $\mathbf{U}$ in formula (20). Hence, $W_\lambda(N)\mathbf{U}_w$ and $\mathbf{U}$ are related via the post-multiplication by an invertible matrix. From this, the conclusion of Lemma 2 follows.

Proof of Corollary 1. For A a matrix and n an integer, denote by $[A]_{\leq n}$ the matrix obtained by zeroing all singular values of rank $> n$ in the SVD of A, and set $[A]_{>n} = A - [A]_{\leq n}$. We successively prove points 1 and 2.

Consider first point 1. Since $\mathcal{H}_p(N)$ satisfies Condition 1, then

$$\left[\mathcal{H}_p^w(N)\mathcal{H}_p^w(N)^T\right]_{>n} = [\mathcal{H}_p(N)]_{>n} \quad \to \quad 0 \tag{61}$$

holds. By the orthogonality property of the SVD, we have

$$\left[\mathcal{H}_p^w(N)\right]_{>n} \mathcal{H}_p^w(N)^T \quad = \quad \left[\mathcal{H}_p^w(N)\mathcal{H}_p^w(N)^T\right]_{>n},$$

whence

$$\left[\mathcal{H}_p^w(N)\right]_{>n} \mathcal{H}_p^w(N)^T \quad \to \quad 0. \tag{62}$$

Matrices $\mathcal{H}_p(N)$ and $\mathcal{H}_p^w(N)$ are related as in Lemma 2 with $W_\lambda(N) = I$ and $W_\rho(N) = \mathcal{H}_p^w(N)$. With this choice for the weights, (62) is exactly (23). On the other hand, since $\mathcal{H}_p(N)$ satisfies Condition 1, then, by Theorem 3, the pair $(C(N), A(N))$ is consistent. Thus, Lemma 2 applies and yields the consistency of $(C_w(N), A_w(N))$.

Consider now point 2. Since $\mathcal{H}_p^w(N)$ satisfies Condition 1, it follows that $\left[\mathcal{H}_p^w(N)\right]_{>n} \to 0$, which implies (61), and thus also (62). Since $\mathcal{H}_p^w(N)$ satisfies Condition 1, then, by Theorem 3, the pair $(C^w(N), A^w(N))$ is consistent. And we conclude again, by a reverse use of Lemma 2.

References

[1] M. Basseville, A. Benveniste, M. Goursat, L. Hermans, L. Mevel, H. Van der Auweraer: "Output-only subspace-based structural identification: from theory to industrial testing practice". *ASME Journal of Dynamic Systems Measurement and Control*, Special Issue on Identification of Mechanical Systems, vol.123, no 4, Dec. 2001, 668-676.

[2] Bauer D., Deistler M. and Scherrer W. On the impact of weighting matrices in subspace algorithms. *in Proceedings of IFAC international symposium on identification*. Santa Barbara, CA. 2000.

[3] Bauer D., M Jansson M.. Analysis of the asymptotic properties of the MOESP type of subspace algorithms. *Automatica*, 36. 497-509. 2000.

[4] Bauer D., Deistler M. and Scherrer W. Consistency and asymptotic Normality of some subspace algorithms without observed inputs. *Automatica*, 35, 1243-1254, 2002.

[5] Bauer D., Ljung L. Some facts about the choice of the weighting matrices in Larimore type of subspace algorithms. *Automatica*. 36, 763-773. 2002.

[6] Benveniste, A. and Fuchs, J-J. Single sample modal identification of a non-stationary stochastic process. *IEEE Trans. on Automatic Control,* AC-30(1), 66-74, Jan. 1985.

[7] Cauberghe B., Guillaume P., Verboven P., and Parloo E. Identification of modal parameters including unmeasured forces and transient effects. *Journal of Sound and Vibration*, 265-3, 609-625, August 2003.

[8] Cauberghe B. Applied frequency domain system identification in the field of experimental and operational modal analysis. PHD Thesis. Vrije Universitet Brussel, Belgium. 2004.

[9] Cauberghe B., Guillaume P. and Pintelon R. Combined Deterministic-Stochastic Frequency Domain Subspace. Submitted for publication.

[10] Chiuso A. and Picci G. Subspace identification by data orthogonalization and model decoupling. *Automatica*, 40, 1689-1704, 2004.

[11] Chiuso A. and Picci G. Asymptotic variance of Subspace methods by data orthogonalization and model decoupling: a comparative analysis *Automatica*, 40, 1705-1717, 2004.

[12] Chiuso A. and Picci G. On the ill conditioning of subspace identification with inputs. *Automatica*, 40, 575-589, 2004.

[13] Deistler M, Peternell K. and Scherrer W. Consistency and relative efficiency of subspace methods. *Automatica*, 31, 1865-1875, 1995.

[14] Guyader A and Mevel L. Covariance driven subspace methods : input/output vs. output-only. *Proceedings of International Modal Analysis conference*, Orlando, February 2003.

[15] Hall P., and Heyde C.C. *Martingale limit theory and its applications.* Academic Press, Inc. (1980).

[16] Larimore W.E. System Identification, reduced order filters and modelling via canonical variate analysis. In *Proceedings of the American Control Conference*, 445-451, San Francisco, June 1983.

[17] Mc Kelvey T. Frequency domain system identification with instrumental based algorithm. In *Proc of the 16th biennal conference on mechanical vibration and noise*, Sacramento, Sept. 1997.

[18] Mc Kelvey T, Akcay H. and Ljung L. Subspace based multivariable system identification from frequency response data. *IEEE Transactions on Automatic Control*. 41-7, 960-978, 1996.

[19] Moustakides G., Benveniste A. Detecting changes in the AR part of a nonstationary ARMA process. *Stochastics,* 16, 137–155. 1986.

[20] Neveu J. *Discrete-parameter martingales.* Elsevier. 1975.

[21] Prevosto M., Olagnon M., Benveniste A., Basseville M., Le Vey G. State-space formulation, a solution to modal parameter estimation. *Journal of Sound and Vibration*. 148, no 2, pp.329-342, 1991.

[22] Van Overschee P., DeMoor B. N4Sid : Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30, 75-93, 1994.

[23] Van Overschee P., DeMoor B. *Subspace identification for linear systems.* Kluwer Academy Publishers. 1996.

[24] Verdult V., Verhaegen M. Subspace Identification of multivariable linear parameter-varying systems. *Automatica*, 38, 805-814, 2002.

[25] Verhaegen M. and Dewilde P. Subspace model identification part 3. Analysis of the ordinary output-error state space model identification algorithm. *International Journal of control*. 55, 555-586. 1993.

[26] Verhaegen M. Identification of the deterministic part of MIMO state space models given in innovation forms from input-output data. *Automatica*, 30, 61-74, 1994.

[27] Viberg M. Subspace-based methods for the identification of linear time-invariant systems. *Automatica*, 31(12), 1835-1851. 1995.

[28] Viberg M., Wahlberg, B. Ottersen B. Analysis of state space system identification methods based on instrumental variables and subspace fitting. *Automatica*, 33, 1603-1616. 1997.